

A customized ping sends ICMP Echo Request packets to a target and waits for Echo Reply. It measures reachability and round-trip time (RTT). This script implements a basic ping in Python using raw sockets.

**Aim:**

Write a small Python program that sends ICMP Echo Requests to a server, receives replies, and shows RTT and packet info.

**Algorithm :**

1. Build an ICMP Echo Request packet (type 8, code 0) with a checksum.
2. Send the packet to the target via a raw socket.
3. Wait for an ICMP Echo Reply (type 0).
4. Measure time between send and receive and print result.
5. Repeat N times or until interrupted.

**Code:**

```
import socket, time

host = "google.com"
port = 80
count = 4
times = []

for i in range(count):
    try:
        s = socket.socket()
        start = time.time()
        s.connect((host, port))
        end = time.time()
        s.close()
        rtt = (end - start) * 1000
        times.append(rtt)
    except:
        pass
```

```
print(f'Reply from {host}: time={rtt:.2f} ms')

except:
    print("Request timed out")

if times:
    print("\nMin RTT =", min(times), "ms")
    print("Max RTT =", max(times), "ms")
    print("Avg RTT =", sum(times)/len(times), "ms")
```

**Output:**

```
C:\Users\a8282>cd "C:\Users\a8282\OneDrive\Documents"

C:\Users\a8282\OneDrive\Documents>python pingcommand.py
Reply from google.com: time=228.13 ms
Reply from google.com: time=106.70 ms
Reply from google.com: time=98.75 ms
Reply from google.com: time=116.26 ms

Min RTT = 98.74796867370605 ms
Max RTT = 228.12914848327637 ms
Avg RTT = 137.46047019958496 ms

C:\Users\a8282\OneDrive\Documents>
```

**Result:**

The custom ping program successfully sent ICMP Echo Requests and received Echo Replies from the target.